

GitHub

Política de nombramiento de ramas

UVUS/[número de issue]-nueva_funcionalidad (palabras separadas por barras bajas y en minúsculas omitiendo artículos y palabras no relevantes)

lucantdel/4-exportacion_censo_csv

IMPORTANTE: Crear ramas SIEMPRE a partir de master.

Política de mensajes de commits

Separación del Título y el Cuerpo:

- Utiliza una línea en blanco entre el título del commit y el cuerpo.
- Limita el título a 50 caracteres y el cuerpo a 72 caracteres por línea.

feat: Agregar nueva funcionalidad

Descripción más detallada de la nueva funcionalidad y cómo se implementa.

Tipos de Commit:

- Utiliza palabras clave en el título para indicar el propósito del commit:
 - feat (Feature):
 - Se utiliza para indicar la adición de una nueva característica o funcionalidad al proyecto.
 - *feat: Agregar funcionalidad de búsqueda por categoría*
 - fix:
 - Indica la corrección de un error o bug en el código.
 - *fix: Corregir manejo incorrecto de excepciones en el formulario de registro*
 - docs:
 - Utilizado cuando se realiza una modificación o adición a la documentación del proyecto.
 - *docs: Actualizar la documentación del API con ejemplos claros*
 - style:
 - Refiere a cambios que no afectan la lógica del código (espacios en blanco, formato, punto y coma, etc.).
 - *style: Alinear código según estándares de estilo*
 - refactor:
 - Indica una modificación en el código que no corrige errores ni añade nuevas características, sino que mejora la estructura o eficiencia del código existente.

- *refactor: Reorganizar la estructura de archivos para mejorar la legibilidad*
- test:
 - Se refiere a la adición o modificación de pruebas unitarias o pruebas de integración.
 - *test: Agregar casos de prueba para la funcionalidad de autenticación*
- chore:
 - Utilizado para cambios en el código que no afectan la lógica de negocio ni las pruebas. Pueden ser tareas de construcción, configuración, etc.
 - *chore: Actualizar dependencias del proyecto*

Título Explicativo:

- El título debe ser conciso pero informativo, describiendo el cambio en términos claros y comprensibles.

Cuerpo Informativo:

- Proporciona detalles adicionales en el cuerpo del commit.
- Explica el porqué del cambio y cualquier información relevante.

Política de Issues y Project

1. Crear Issue con nombre de la funcionalidad.
2. Asignarte a ti mismo.
3. Asignar el milestone correspondiente.
4. Mover la issue al estado correspondiente (inicialmente Todo o In Progress).
5. Una vez implementada la funcionalidad, mover a In Review.
6. Crear una PR y asignar un compañero para su revisión.
7. El compañero asignado revisa, comenta, mergea, mueve la issue a Done y la cierra.